IN THE SPECIFICATION:

Please replace the paragraph extending from page 9, line 17 through page 10, line 8 with the following, in which a comma is inserted in line 4 and "of" is changed to "if" on line 13:

Refer now to FIGURE 5, illustrating event driven graphic display process 500 in accordance with the principles of the present invention. Process 500 may be used with an embodiment of graphical image data in accordance with the format described hereinabove in conjunction with FIGURES 4.1 and 4.2. Note that, as used herein and described further below, a graphical image may constitute a portion of a composite image. In step 501, a first graphical image is processed. In processing the graphical image in step 501, control and image data may be read from the associated control block and rendering block, respectively, and the image data in the rendering block decoded to generate graphic data which may be rendered on a display device, such as display 338, FIGURE 3. The graphical data may be rendered on the display in response to control data in the corresponding control block. Note that control data may include an input flag such as input flag 410, FIGURE 4.2. As previously described, a control block may include field containing a delay time value, if, in step 502, the delay time value is not zero, the graphics file is processed as an animation, step 518. Otherwise, in step 503 it is determined [[of]] if the input flag, such as input flag 410, FIGURE 4.2, is set. If, in step 503, the input flag is set, process 500 proceeds to step 504, to process graphical control images in association with events. Otherwise, if the input flag is reset, that is negated (in its logically false state), then, in step 518, the graphics file is processed as an animation using delay time information, as described, for example, in GIF 89a.

Please replace the paragraph extending from page 10, line 20 through page 11 line 14 with the following, in which duplicate wording is deleted:

On the receipt of an event targeted to process 500 as defined by the event handler, step 504 breaks out of the loop. In step 505, an event identifier (ID) is received. The event ID may be a value passed to process 500 by the event handler. In step 506, the next image control block in sequence in the graphical image is processed. Steps 506, 508 and 510 form a loop over graphical images that may be applied to the images already displayed, in particular to the image displayed in step 501. While, in step 508, the image ID, for example the value in image ID field 408 in an embodiment of an image control block in accordance with FIGURE 4.2 described hereinabove, corresponds to the event ID value received in step 505, ~~the corresponding~~ the corresponding graphic rendering block is decoded and the graphic rendered on the display device, step 510. Process 500 then returns to step 506 to close the aforementioned loop. Note that the graphical data in the blocks rendered in step 510 may constitute only such data as is necessary to update the first image, already displayed on the screen. In other words, the graphical data may include differential information between the first image and the updated image whereby the updated image is formed as a composite. Additionally, in an embodiment in accordance with the GIF Specification, the differential data may be aggregated to form the composite by setting the value in disposal method flag 416, FIGURE 4.2 corresponding to "do not dispose," which value may be 1. In this way, a button or similar control background may be inverted, for example, to depict that the button or similar control has been selected, without having to store and retrieve a separate image file to display the selected control. The image data is decoded and rendered in the same fashion as the processing of the first image in the file, discussed hereinabove in conjunction with step 502.